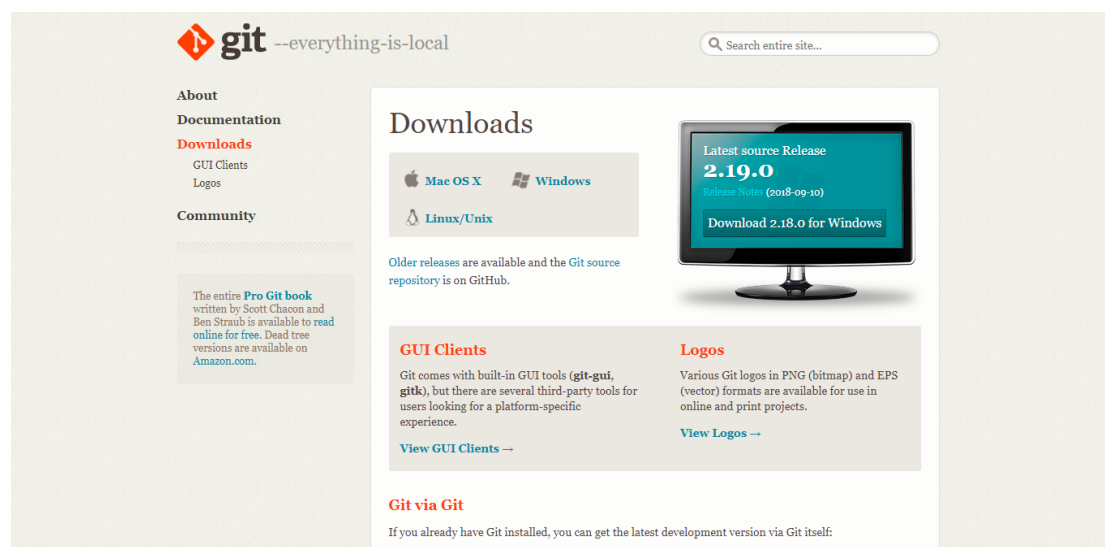
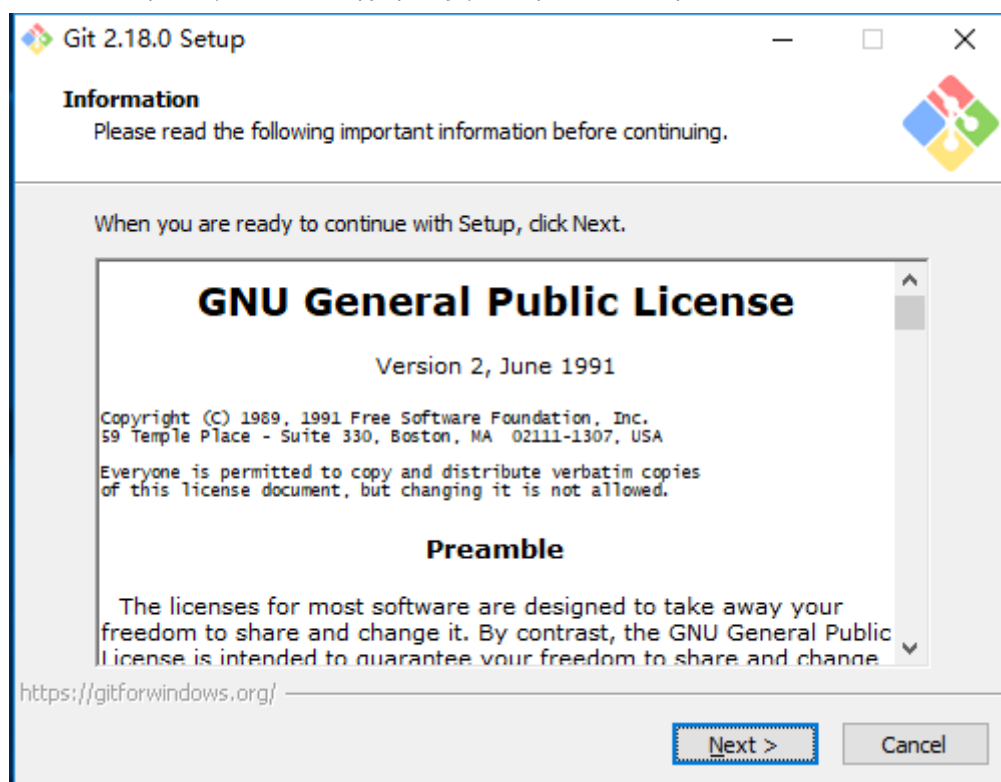


一、Git 安装

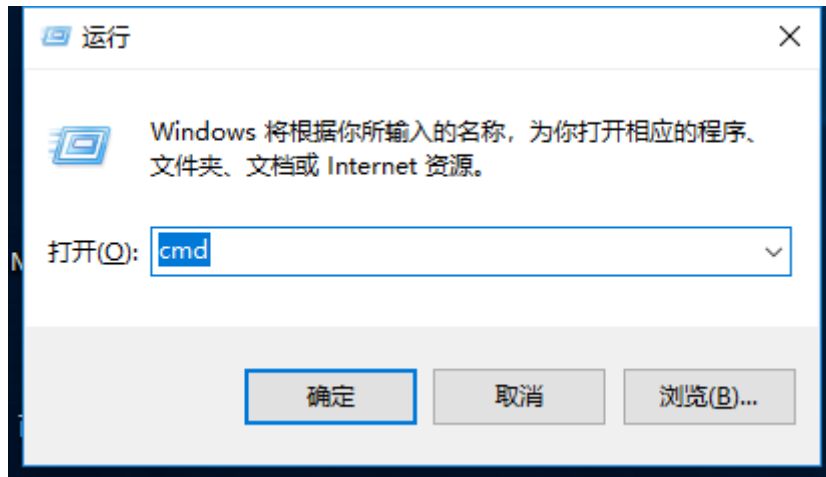
下载地址：<https://git-scm.com/downloads>，根据自己系统选择相应版本。



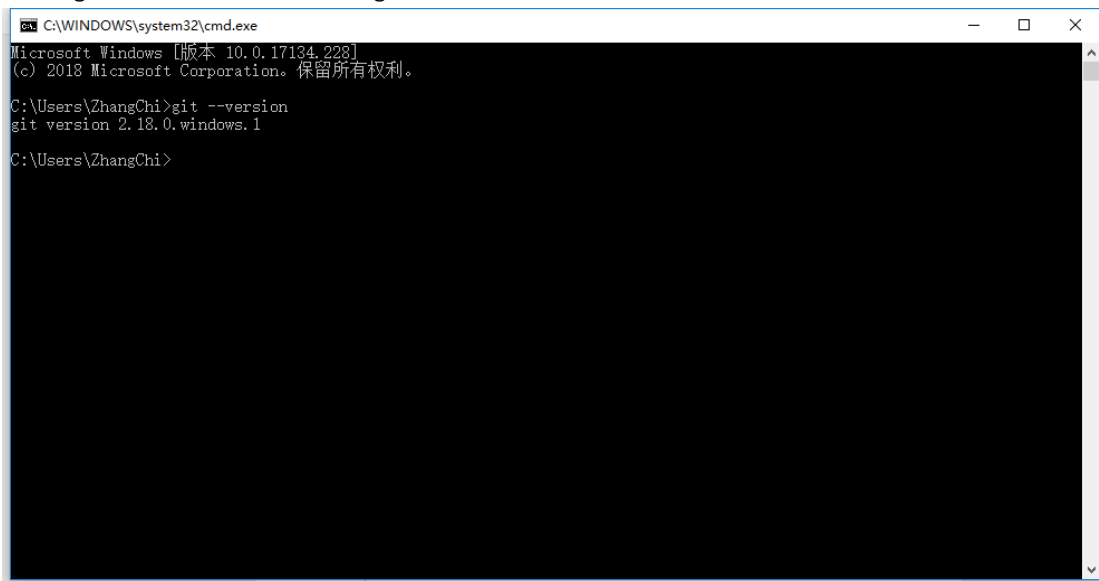
下载完成后双击打开，如果没有特殊要求，选择默认设置即可。



WIN+R 打开运行，输入 cmd 打开命令行。

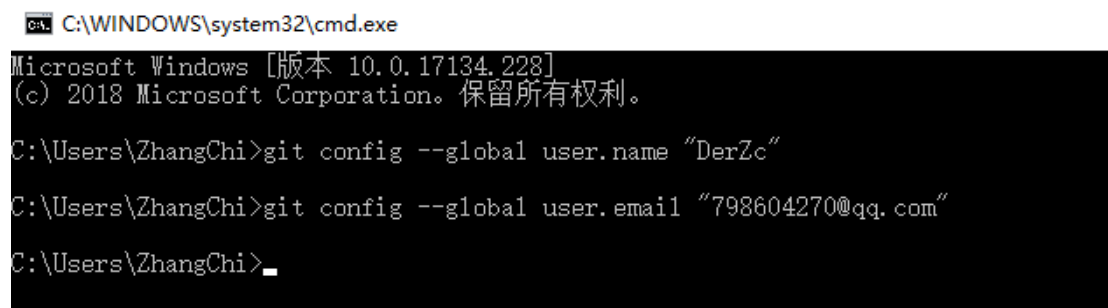


输入 `git --version` , 返回当前 git 版本信息即为成功。



设置自己的信息, 填入用户名和邮箱, Github 注册在下一步骤中。

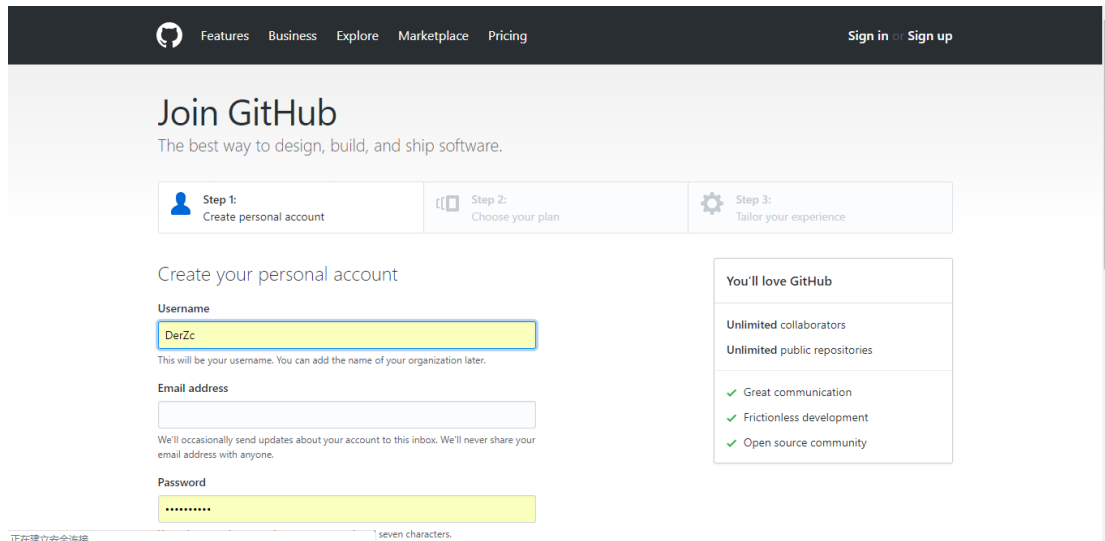
```
git config --global user.name "Your Name"  
git config --global user.email "email@example.com"
```



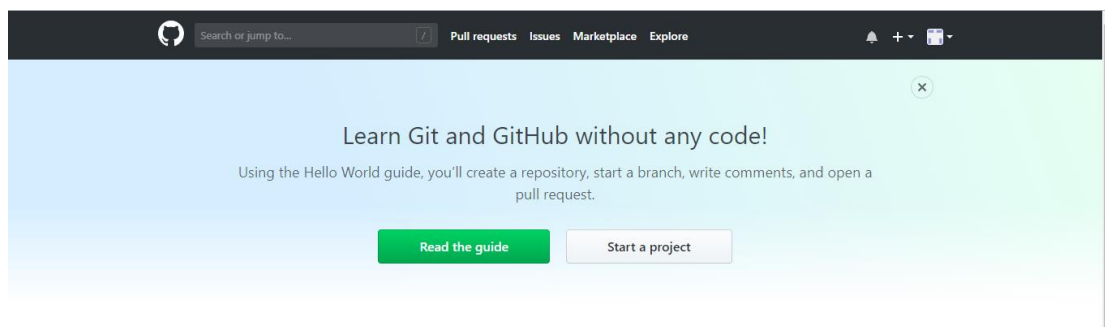
二、Github 使用教程

网址为 <https://github.com/>

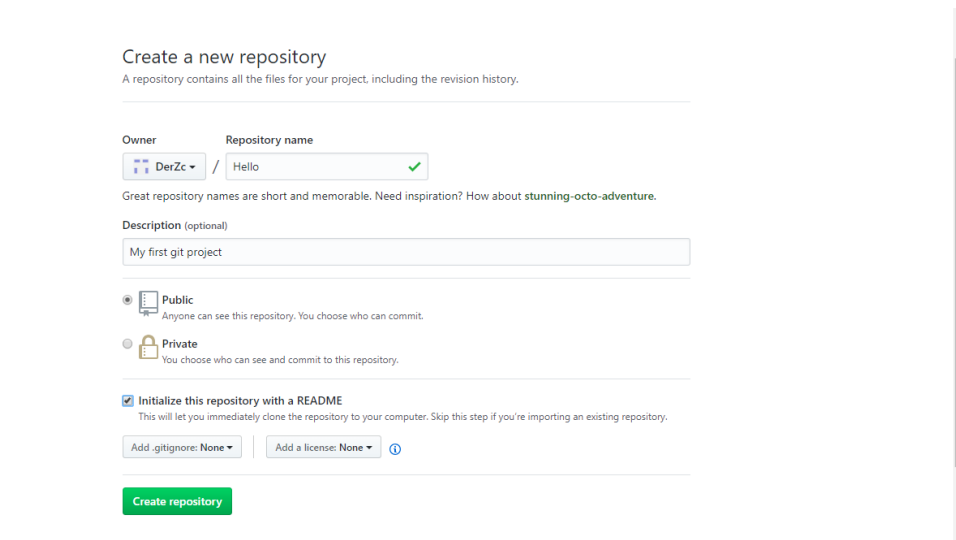
打开后点击 Sign Up 进行账号注册。



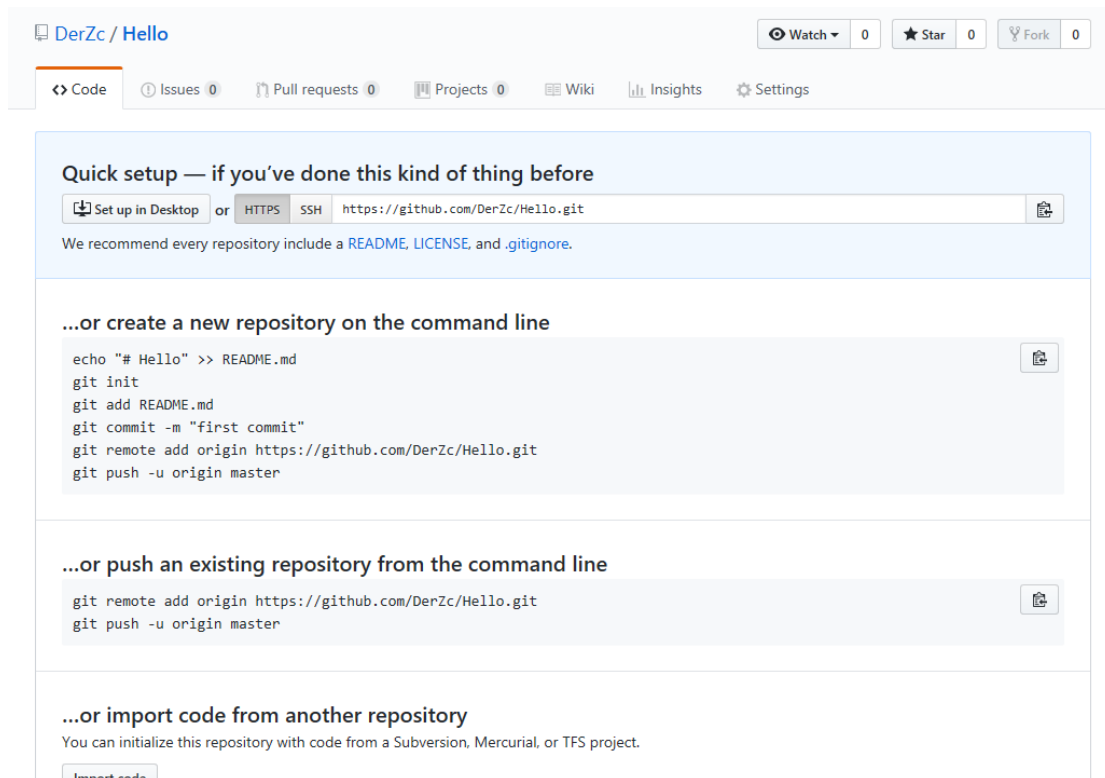
注册完成后进行登陆，进入自己主页，点击 Start a project 创建一个新的代码仓库。



填写仓库名称，仓库介绍，勾选权限，根据下方提示选择是否创建 README 文件，点击 Create。



出现如下界面，远程仓库创建完成。可根据图中所展示的命令行进行操作，建立本地仓库与 Github 仓库的连接，也可继续阅读本教程。



打开命令行工具，进入自己 Android 项目所在文件夹。实例中为 D:\Git_test
输入 `git init` 初始化本地仓库

```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [版本 10.0.17134.228]
(c) 2018 Microsoft Corporation. 保留所有权利。

C:\Users\ZhangChi>D:
D:\>cd Git_test
D:\Git_test>git init
Initialized empty Git repository in D:/Git_test/.git/
```

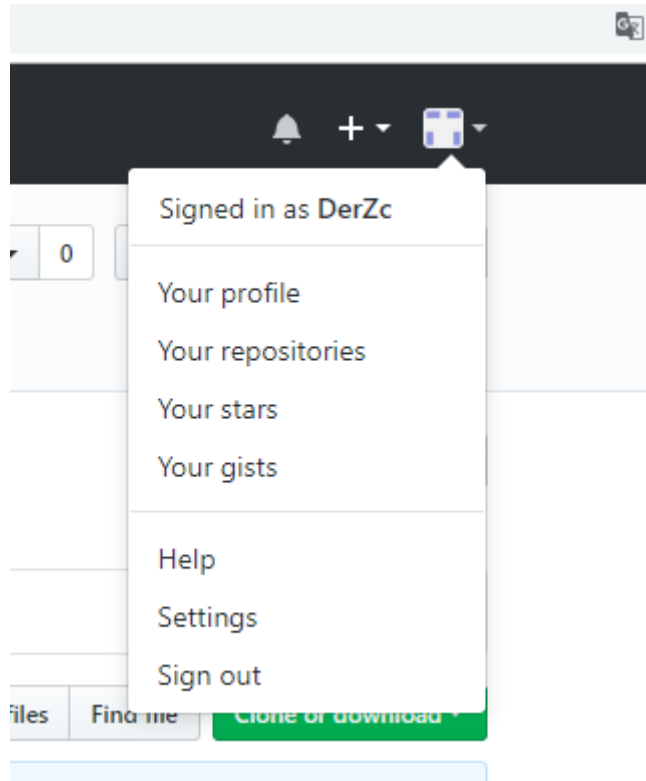
输入 `git add .` 将目录下所有文件添加到仓库中。
输入 `git commit`，将文件提交到仓库中。命令后可跟 `-m` 参数，表示本次提交的注释内容。

```
D:\Git_test>git add .
D:\Git_test>git commit -m "The first commit"
[master (root-commit) 5f18efb] The first commit
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 Code.cpp
```

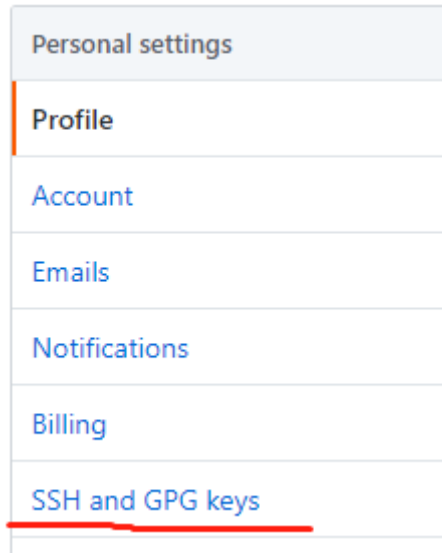
本地 Git 仓库与 Github 仓库之间的传输通过 SSH 加密，需进行设置。
检查 C 盘用户目录下有没有 `.ssh` 目录，没有的话通过一下命令创建，
`ssh-keygen -t rsa -C youremail@example.com`
输入后所有选择默认即可，此时 C 盘的用户目录下会产生 `.ssh` 目录，目录中有 `id_rsa` 和 `id_rsa.pub` 两个文件。用记事本打开 `id_rsa.pub`，复制里面内容。

名称	修改日期	类型	大小
id_rsa	2018/9/12 19:46	文件	2 KB
id_rsa.pub	2018/9/12 19:46	Microsoft Publis...	1 KB

点击 Github 网站上右上角图标，点击 Settings，



找到右侧 SSH and GPG keys 选项，点击，



点击右侧绿色图标 New SSH key

SSH keys

New SSH key

There are no SSH keys associated with your account.

Check out our guide to [generating SSH keys](#) or troubleshoot [common SSH Problems](#).

Title 中可随意填写，Key 中填写刚才复制内容。

SSH keys / Add new

Title

title

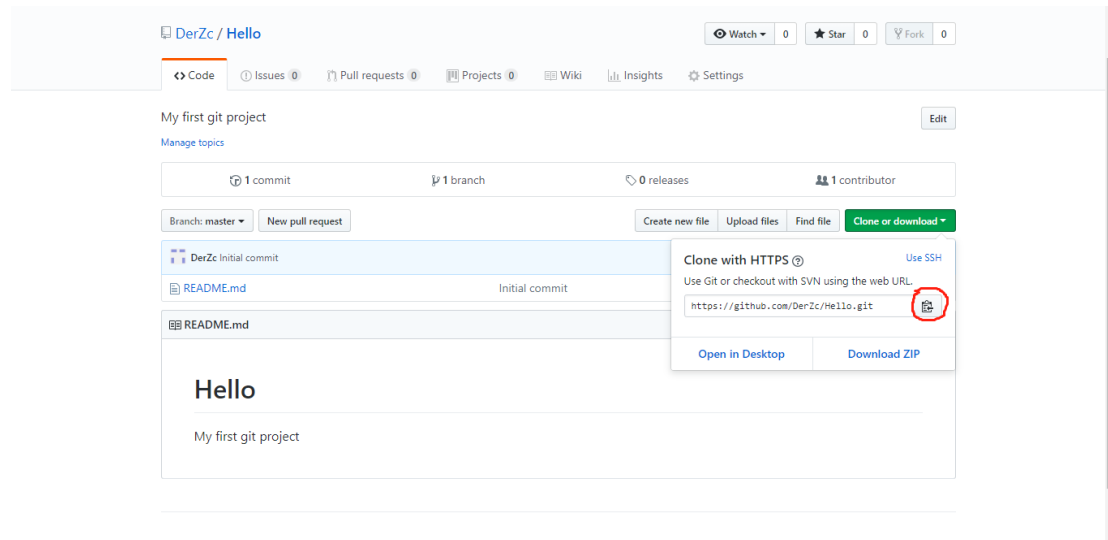
Key

```
ssh-rsa
AAAAAB3NzaC1yc2EAAAADAQABAAQAC/4O093hq5o3WCD3p5PFHjjGL2CSceYsxSqEInCnyjUoEYQUVRB/olVfgZ1p4
dD+vyNx8rg0LfyRNcW948ZocIKtLXRC2wk0rNHceObHWkpMVy6eZ+/FhKqjKKqzITPEmPZbrwlpCChL9VzesD8fSWU
YJYh0WZXm7HNY7ITBW+iBQw/buSadxNPcw4sRnUCXE6CBcXhmHimVkCQZPSXms5bCKsZW1rPMMmg6qkv0dS0W
qx3e9gNT2TH9xNEGBOWfgkS/1+UeA0tB7tNvaTASBikf0X0ujUMLIBmH7nNR0oBwe85GyHLo/TmrisKVQeJU7x9bbAlje
z4zmpIG8g/J 798604270@qq.com
```

Add SSH key

点击下方绿色按钮 Add SSH key，加密完成。

回到开始创建的 Github 仓库，点击右侧绿色按钮，复制远程仓库地址。



The screenshot shows a GitHub repository page for 'DerZc / Hello'. The repository has 1 commit, 1 branch, 0 releases, and 1 contributor. The 'Clone or download' button is highlighted with a red circle. The repository content shows a README.md file with the text 'Hello' and 'My first git project'.

在本地仓库的命令行中输入以下命令，建立本地仓库与远程仓库连接。

git remote add origin <https://github.com/DerZc/Hello.git> (后面为自己仓库地址)

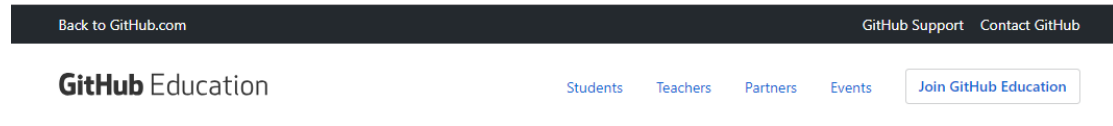
输入以下命令，将本地仓库代码提交至 Github 仓库。

```
git push -u origin master
```

会提示输入用户名和密码，填写正确即可。

DerZc The first commit	Latest commit 5f18efb 38 minutes ago
Code.cpp	The first commit 38 minutes ago

Github 学生认证(用 edu 邮箱认证), 可免费创建私有项目:
<https://education.github.com/>

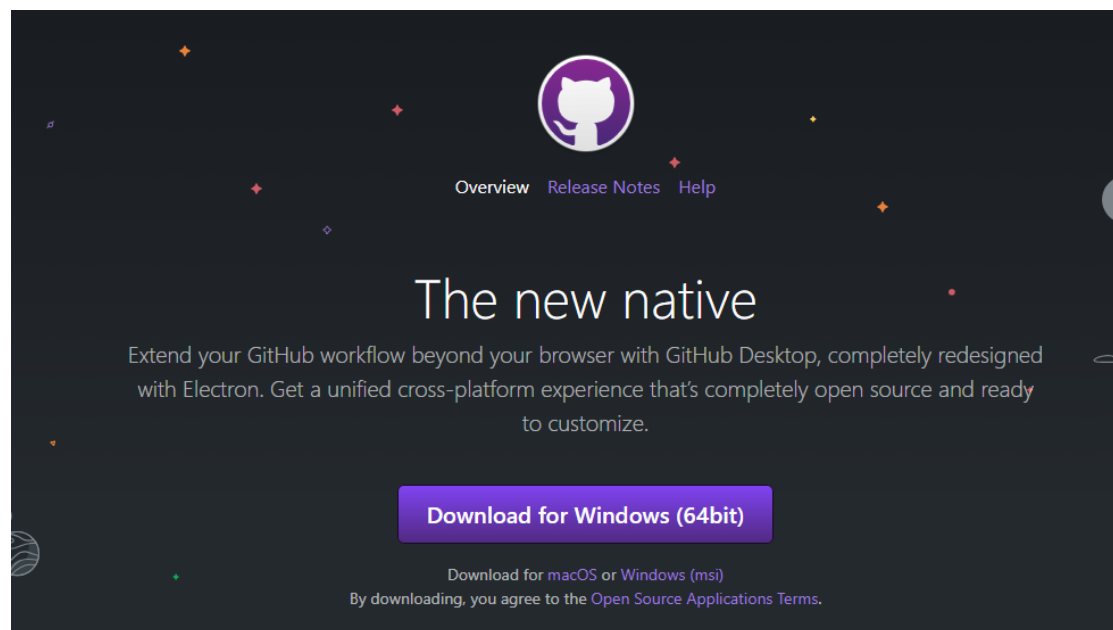


Real-world tools, engaged students

GitHub Education helps students, teachers, and schools access the tools and events they need to shape the next generation of software development.

Github 客户端 (Windows, MacOS):
<https://desktop.github.com/>

可视化地进行 commit,push,fetch,pull 等操作, 对所修改的文件进行查看和对比 (根据个人喜好选择使用)。



三、Gradle 安装与使用

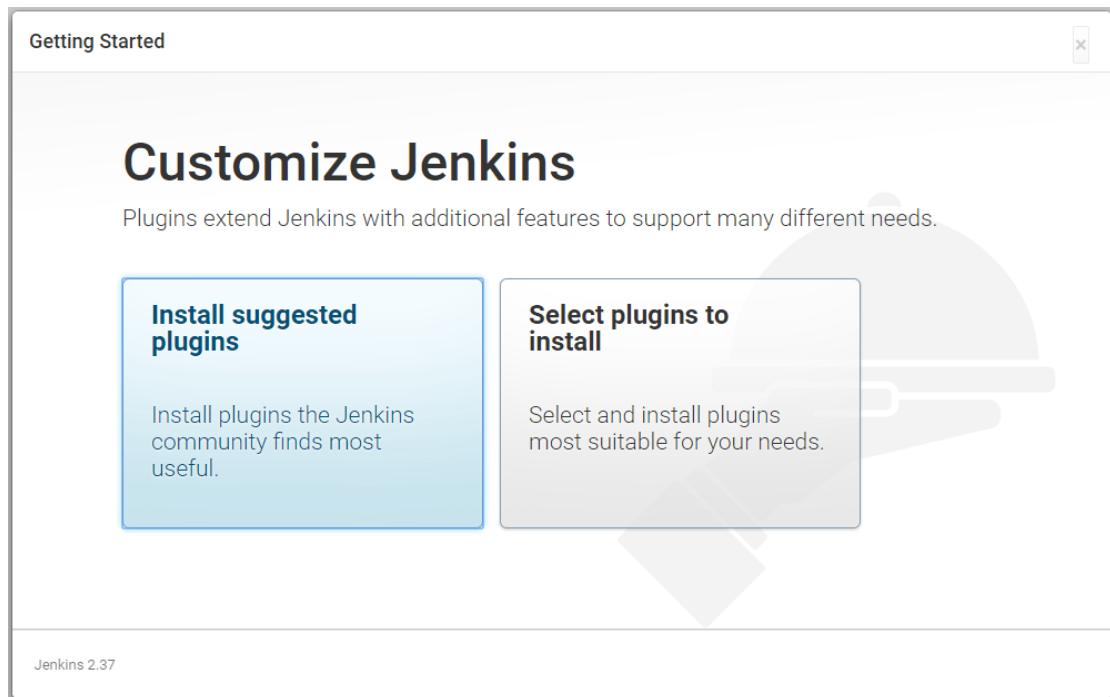
因 Android Studio 中集成了 Gradle 工具, 则不再进行额外安装。

四、Jenkins 安装与使用

前提: Git、JDK、SDK、Android Studio 均已安装成功。

下载地址: <https://jenkins.io/download/> 根据电脑版本下载, 选择 LTS 即可。

下载后双击进行安装。按照提示进行即可。



插件安装选择推荐安装即可，如有其他需求可自己选择安装，后期也可选择安装其他插件。插件安装完成后进行系统设置。点击右侧系统管理。



选择全局工具配置。

管理Jenkins



系统设置
全局设置&路径



全局安全配置
Jenkins安全，定义谁可以访问或使用系统。



凭据配置
配置凭据的提供者和类型



全局工具配置
工具配置，包括它们的位置和自动安装器

全局工具配置



读取设置
放弃当前内存中所有的设置信息并从配置文件中重!

JDK 安装中别名内填写本机 JDK 地址，下方从 Java.sun.com 安装可不进行操作。

JDK

JDK 安装

新增 JDK

JDK

别名

自动安装 ?

从 java.sun.com 安装 ?

版本

我同意 Java SE Development Kit 的许可协议

Installing JDK requires Oracle account. Please enter your username/password

新增安装

Git 安装可不进行操作。

点击下方 Gradle 安装，填写本机 Gradle 地址，如未额外安装 Gradle，可填写 Android Studio 内集成 Gradle 地址，地址如图所示。虽 Jenkins 中已有 Gradle，为保持同本地编译环境 Gradle 版本相同，推荐填写本地 Gradle 地址，此处可添加多个版本 Gradle。

Gradle

Gradle 安装

新增 Gradle

Gradle

name E:\Java\AndroidStudio\gradle\gradle-4.4\bin

自动安装

Install from Gradle.org

版本 Gradle 4.10

新增安装

新增 Gradle

系统下Gradle 安装列表

删除安装

删除 Gradle

设置完成后点击保存。

在系统管理->系统设置->全局属性版块勾选上 Environment variables 选项, 添加 SDK 路径。

全局属性

工具位置

环境变量

键值对列表

键 ANDROID_HOME

值 E:\Java\SDKManager

删除

键 JAVA_HOME

值 E:\Java\JDK

删除

增加

如后期实验过程中出现 JAVA_HOME 路径错误, 也可将 JDK 路径填写在此处。

然后返回主页, 新建一个任务。

输入一个任务名称

Hello_Android

» 必填项

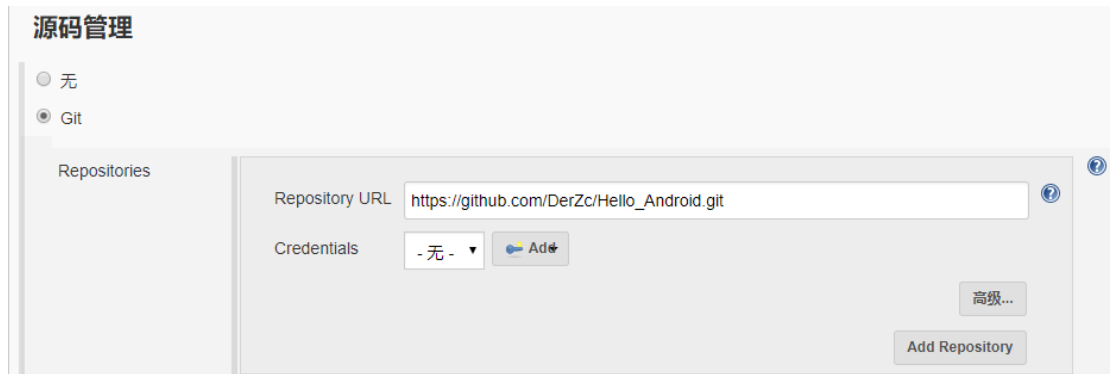
构建一个自由风格的软件项目

这是Jenkins的主要功能.Jenkins将会结合任何SCM和任何构建系统来构建

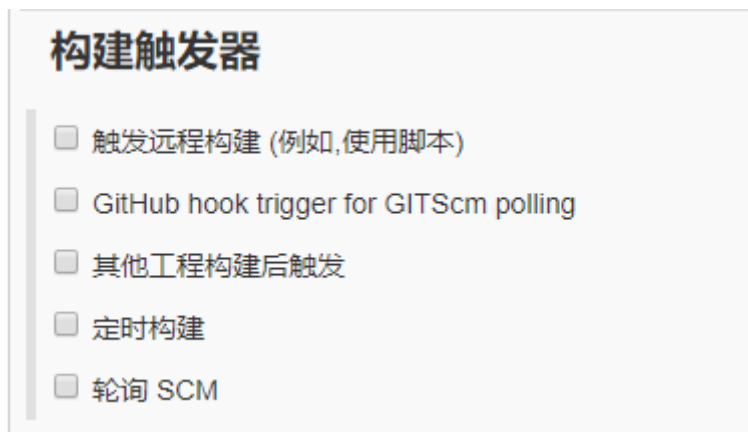
流水线

选择构建一个自由风格的软件项目。

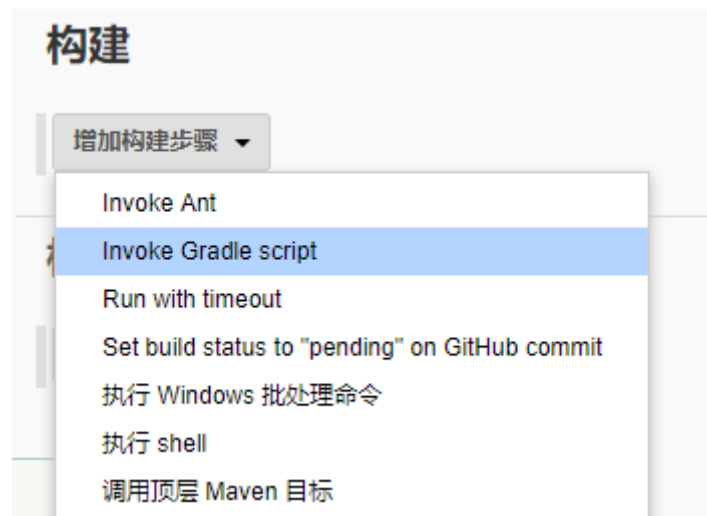
源码管理中选择 Git, 并填写自己项目的 Github 地址。



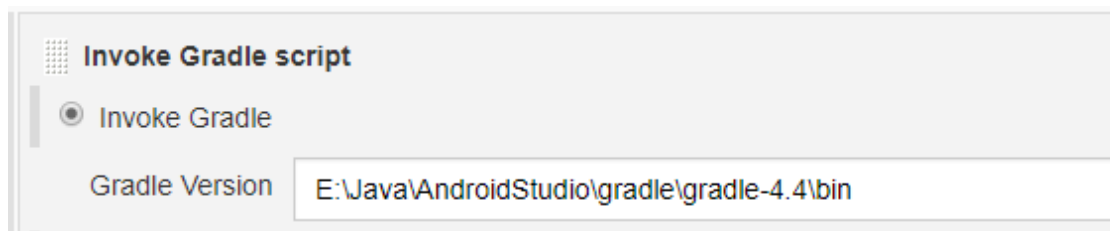
构建触发器可自由选择，如不选择，则每次都由手动构建。



增加构建步骤中选择 Invoke Gradle script



Gradle Version 中选择本机中 Gradle 版本。



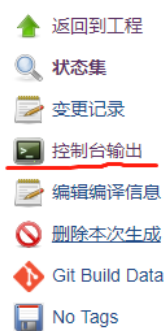
配置完成后点击左侧立即构建，则可对项目进行一次构建。



构建情况如下：



蓝色表示成功，红色表示失败。



构建 #1 (2018-9-12 20:49:42)



No changes.



启动用户 [ZC](#)



Revision: [36f409cc70aeb077f63f94fdf9ef31fe77ad0df4](#)

• [refs/remotes/origin/master](#)

点击构建，选择左侧的控制台输出，可查看构建信息。

控制台输出

由用户 ZC 启动

构建中 在工作空间 E:\Jenkins\workspace\Hello_Android 中

没有接受许可之前不能够自动安装.

没有接受许可之前不能够自动安装.

```
> git.exe rev-parse --is-inside-work-tree # timeout=10
Fetching changes from the remote Git repository
> git.exe config remote.origin.url https://github.com/DerZc/Hello_Android.git # timeout=10
Fetching upstream changes from https://github.com/DerZc/Hello_Android.git
> git.exe --version # timeout=10
> git.exe fetch --tags --progress https://github.com/DerZc/Hello_Android.git +refs/heads/*:refs/remotes/origin/*
> git.exe rev-parse "refs/remotes/origin/master^{commit}" # timeout=10
> git.exe rev-parse "refs/remotes/origin/origin/master^{commit}" # timeout=10
没有接受许可之前不能够自动安装.
Checking out Revision 36f409cc70aeb077f63f94fdf9ef31fe77ad0df4 (refs/remotes/origin/master)
> git.exe config core.sparsecheckout # timeout=10
> git.exe checkout -f 36f409cc70aeb077f63f94fdf9ef31fe77ad0df4
Commit message: "change gradle.properties"
First time build. Skipping changelog.
[Gradle] - Launching build.
没有接受许可之前不能够自动安装.
[Hello_Android] $ cmd.exe /C "E:\Jenkins\tools\hudson.plugins.gradle.GradleInstallation\E_Java_AndroidStudio_gradle_gradle
Starting a Gradle Daemon, 1 incompatible and 1 stopped Daemons could not be reused, use --status for details

> Configure project :app
Observed package id 'build-tools;28.0.0-rc2' in inconsistent location 'E:\Java\SDKManager\build-tools\28.0.0-preview' (E:
rc2')

> Task :help

Welcome to Gradle 4.10.

To run a build, run gradle <task> ...
```

构建完成后，可添加插件进行其他代码质量检查操作、构建失败通知等。

构建后操作

Publish Checkstyle analysis results ✕ ?

Checkstyle results

[Fileset includes](#) setting that specifies the generated raw CheckStyle XML report files, such as `**/checkstyle-result.xml`. Basedir of the fileset is [the workspace root](#). If no value is set, then the default `**/checkstyle-result.xml` is used. Be sure not to include any non-report files into this pattern.

高级...

E-mail Notification ✕ ?

Recipients

Whitespace-separated list of recipient addresses. May reference build parameters like `$PARAM`. E-mail will be sent when a build fails, becomes unstable or returns to stable.

每次不稳定的构建都发送邮件通知

单独发送邮件给对构建造成不良影响的责任人

增加构建后操作步骤 ▾